WHAT IS CLAIMED IS:

1. A method invoking applications associated with a first software environment in a second software environment while migrating software applications from the first software environment to the second software environment, the method comprising:

initiating a request message for invoking an application by a client according to a message bus based architecture; and

utilizing an Interface Definition Language (IDL) interface based on the request message, the IDL interface using input and output buffers for information exchange conforming to the second software environment,

wherein the request message is passed through the input buffer to a predetermined server hosting the application and reply message data returned therefrom is passed through the output buffer.

2. The method of claim 1 wherein the utilizing further includes creating an input buffer for receiving one or more messages from the client and an output buffer for receiving the reply message data from the predetermined server.

3. The method of claim 1 wherein the utilizing further includes:

identifying the server hosting the application;

initiating a synchronized communication session with the identified server, the synchronized communication session passing the request message to the identified server;

processing the message by the identified server for invoking the application; and

returning reply message data associated with the invoked application from the server.

4. The method of claim 3 wherein the initiating a synchronized communication session further includes:

locating a server object to receive the request message;

placing the request message in the input buffer;

dispatching the message to the application; and

placing reply message data in the output buffer when the application responses.

5. The method of claim 4 wherein the placing reply message data further includes accumulating the reply message data in a reply buffer before copying same to the output buffer.

6. The method of claim 1 wherein the first software environment is TIBCO RV environment

7. The method of claim 1 wherein the second software environment is a synchronized communication session based environment.

8. The method of claim 7 wherein the second software environment is a COBRA environment.

9. A computer program for invoking applications associating with a TIBCO RV environment in a CORBA environment while migrating software applications from the TIBCO environment to the CORBA environment, the program comprising instructions for:

initiating a request message for invoking an application by a client according to a message bus based architecture; and

utilizing an Interface Definition Language (IDL) interface based on the request message, the IDL interface using an input and output buffers for information exchange conforming to the second software environment,

wherein the request message is passed through the input buffer to a predetermined server hosting the application and reply message data returned therefrom is passed through the output buffer.

10. The program of claim 9 wherein the instructions for utilizing further includes instructions for creating an input buffer for receiving one or more messages from the client and an output buffer for receiving the reply message data from the predetermined server.

11. The program of claim 9 wherein the instructions for utilizing further includes instructions for:

identifying the server hosting the application;

initiating a synchronized communication session with the identified server, the synchronized communication session passing the request message to the identified server;

processing the message by the identified server for invoking the application; and

returning reply message data associated with the invoked application from the server.

12. The program of claim 11 wherein the instructions for initiating a synchronized communication session further includes instructions for:

locating a server object to receive the request message;

placing the request message in the input buffer;

dispatching the message to the application; and

placing reply message data in the output buffer when the application responses.

13. The program of claim 12 wherein the instructions for placing reply message data further includes instructions for accumulating the reply message data in a reply buffer before copying same to the output buffer.

14. A method for invoking applications while migrating software applications from a first middleware software environment to a second middleware software environment, the method comprising:

initiating a request message for invoking an application by a client in the CORBA software environment;

sending the request message to an Interface Definition Language (IDL) interface;

marshaling the message into a message object;

establishing a communication session between the IDL interface and a CORBA middleware;

locating a server object to receive the message object;

creating input and output buffers for the established communication session;

placing the request message in the input buffer after extracting same from the message object;

passing the request message in the input buffer to the application hosted by a server;

processing the message by the application;

accumulating reply message data associated with the invoked application from the server to a reply buffer;

copying the reply message data from the reply buffer to the output buffer;

marshaling the reply message data in the output buffer by an IDL skeleton into one or more reply message objects;

sending the marshaled reply message objects to the IDL interface;

unmarshaling the reply message objects; and

conveying the unmarshaled reply message data to the client.

15. A method for operating applications while migrating software applications from a TIBCO software environment to a CORBA software environment, the method comprising:

receiving a request message from a client for operating an application, the request message conforming to TIBCO protocol;

marshaling the message into a message object through a universal application invoking Interface Definition Language (IDL) stub;

conducting a communication session with a CORBA object request broker for locating a server object to identify the application for processing the message object;

receiving reply message data from the application; and

sending the reply message to the client.

16. The method of claim 15 wherein the conducting further includes creating input and output buffers;

placing the request message in the input buffer after extracting same from the message object;

processing the message by the application;

accumulating reply message data associated with the invoked application in a reply buffer; and

copying the reply message data from the reply buffer to the output buffer.

17. The method of claim 16 wherein the conducting further includes:

marshaling the reply message data into one or more reply message objects; and

sending the marshaled reply message objects from an IDL skeleton to the CORBA object request broker.

18. The method of claim 17 further includes:

unmarshaling the reply message objects by the IDL stub; and

conveying the unmarshaled reply message data to the client.